intel®

AP BRIEF

December 1982

The Designer's Guide to iRAMs

Order Number: 210860-001

Additional copies of this manual or other Intel literature may be obtained from:

Literature Department
Intel Corporation
3065 Bowers Avenue
Santa Clara, CA 95051

THE DESIGNER'S GUIDE TO iRAMs

The iRAM is the first of a new generation of VLSI memories; a complete dynamic RAM system on a chip. It combines the advantages of the simple static RAM interface with the high density and low power dissipation of the more economical dynamic RAM. While extraordinarily complex internally (more than 150,000 active elements), the external interface of the iRAM is just slightly different from the interface of the 2K × 8 static RAMs that you have used before. The iRAM can sit in a 28-pin Universal Site. Designs based on the 2186 are fully compatible with EPROMs, EEPROMs, and 8K × 8 static RAMs. This is an important consideration when designing for compatibility with multiple vendors. There are three differences between the iRAM and SRAM interface. These are described below.

This guide is intended to show you how to use the iRAM. It contains a functional summary of the iRAM, and several simple microprocessor and microcontroller application examples. The enclosed access time matrix matches the access speeds of the iRAM to the operating speed of any Intel microprocessor. For more detailed design information, please consult Intel Application Note 132 "Designing Memory Systems with the 8K × 8 iRAM".

FUNCTIONAL DESCRIPTION

Just like a static RAM (or EPROM), access to the iRAM is initiated by activating the Chip Enable control signal ($\overline{CE}$). The bar over $\overline{CE}$ indicates that it is an active low signal. Activating $\overline{CE}$ latches the valid external addresses from the system bus into the RAM. In simple systems, the iRAM's address latches can eliminate the need to demultiplex the address/data bus from the processor.

Because the leading (falling) edge of $\overline{CE}$ starts the internal sequencing of a memory cycle, a transition on the $\overline{CE}$ control signal must be glitch-free. Any spurious transitions of $\overline{CE}$ may inadvertantly select the iRAM at the wrong time, resulting in a loss of data or worse. Figure 1 shows a simple circuit that ensures a clean transition for $\overline{CE}$.

Once $\overline{CE}$ has been activated, the user may select one of three different cycles; a Read cycle, a Write cycle or a False Memory cycle. For a Read cycle, the read control line, called $\overline{OE}$ (Output Enable), is activated. Data will remain valid

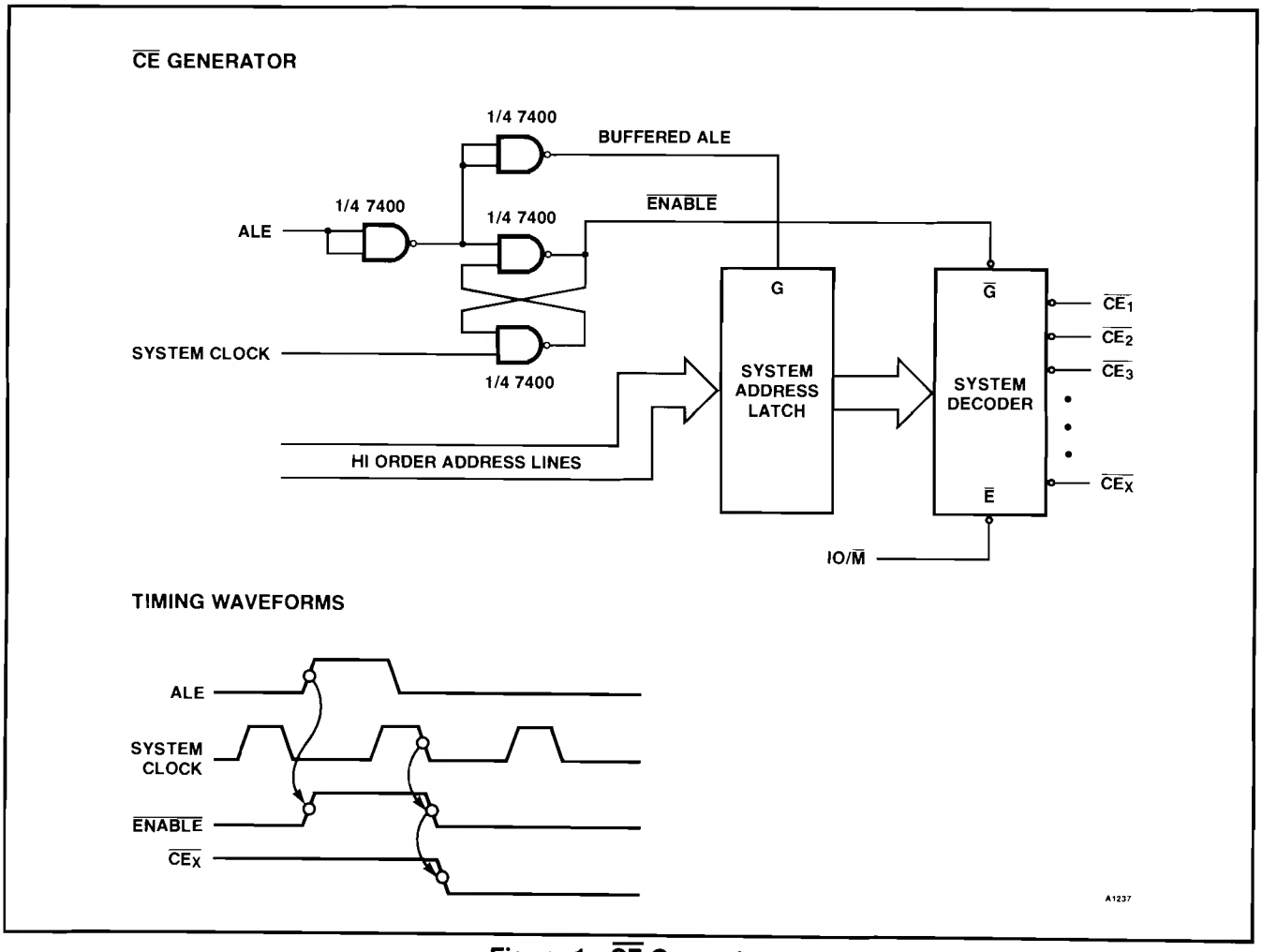


Figure 1. $\overline{CE}$ Generator

at the RAM outputs as long as $\overline{OE}$ is held active — regardless of the state of $\overline{CE}$. $\overline{OE}$ must return to the inactive state prior to the next occurence of $\overline{CE}$.

For a Write cycle, the write control line, called $\overline{WE}$ (Write Enable), is activated after $\overline{CE}$ is enabled. The only requirement for a successful Write cycle is that data be valid at the data inputs prior to $\overline{WE}$ going active. Some older processors, and the iAPX 86/88 in minimum mode, require the addition of a single flip-flop to delay $\overline{WE}$ going active until data is valid. Figure 2 illustrates an example of this circuit and the associated timing waveforms. Finally, $\overline{WE}$ must return to the inactive state prior to the next occurence of $\overline{CE}$. Note that $\overline{OE}$ and $\overline{WE}$ may not both go active during the same cycle.

A False Memory cycle (FMC) occurs whenever $\overline{CE}$ is activated and neither $\overline{OE}$ nor $\overline{WE}$ go active. Addresses must be valid at the iRAM prior to $\overline{CE}$ going active for an FMC. The designer should be aware that some unique timing requirements exist for FMCs.

Because internal refresh may be occurring at any time, a simple handshake procedure is used to notify the processor of a delay in the cycle. If $\overline{CE}$ arrives while the iRAM is in the midst of a refresh cycle, the ready output line (RDY) will go inactive low. RDY is usually used to generate WAIT states, and several RDY lines can be "OR'd" together at the system level. When both the internal refresh cycle and the requested external access cycle are complete, RDY is released and the processor finishes the transfer.

An alternate version of the iRAM gives the designer complete control over access and refresh cycles. On the 2187 iRAM, the RDY output is replaced by the refresh enable input ($\overline{REFEN}$). To initiate a refresh cycle, $\overline{REFEN}$ is activated. An internal refresh row address counter provides the refresh address. Note that $\overline{CE}$ and $\overline{REFEN}$ may not both go active during the same cycle. This synchronous iRAM is especially suited for use with microcontrollers that can not accept a ready handshake line.

In summary, there are three key interface requirements that the iRAM designer must address:

- $\overline{CE}$ input must be glitch free
- $\overline{WE}$ input may have to be delayed until data is valid
- RDY output is used to request a WAIT state during refresh/access overlap.

Some examples of how to use the iRAM are shown on the following pages. A microcontroller application and two different microprocessor systems are outlined, complete with their major interface elements. The timing charts match the operating speed of your processor with the appropriate iRAM access time.

A list of the iRAM literature that is available from Intel is included in this guide. You can get this literature, and answers to any questions about the availability and pricing of the iRAM from your local Intel sales office.

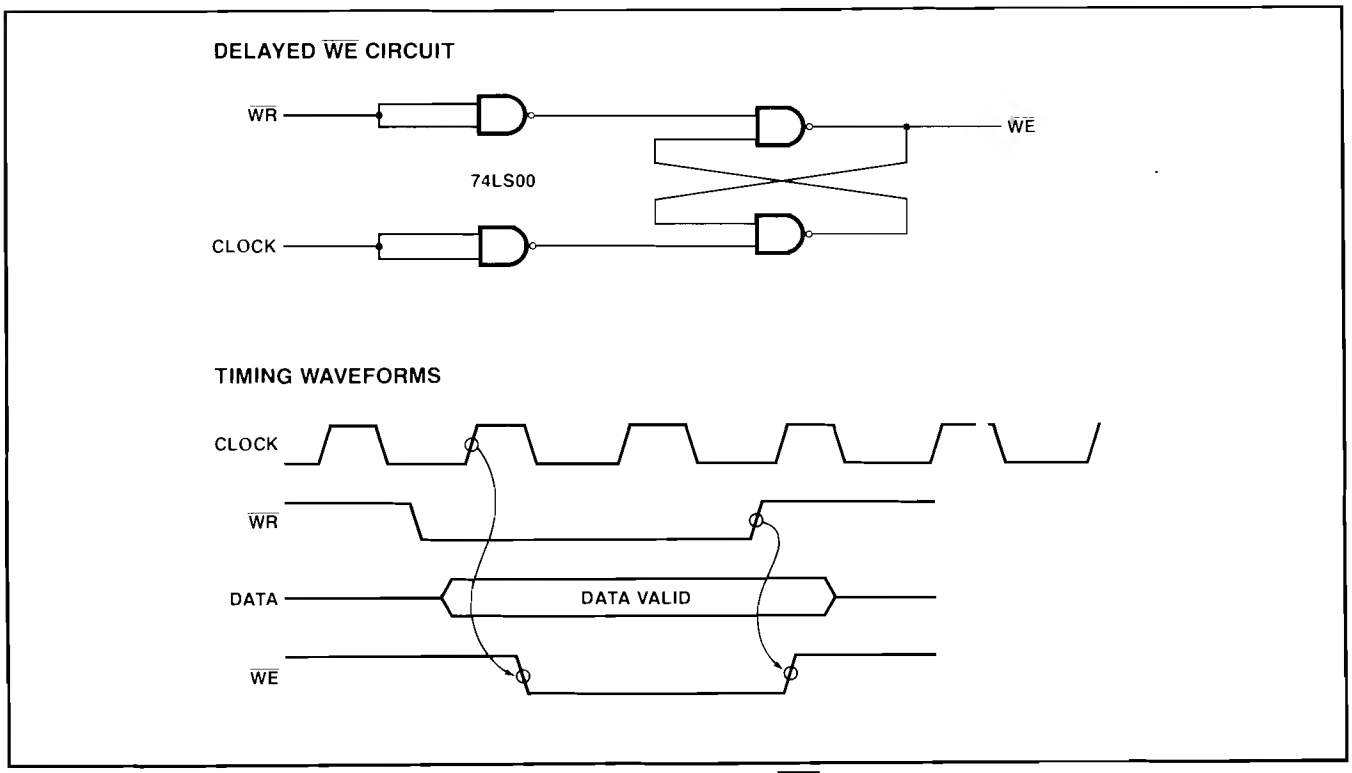
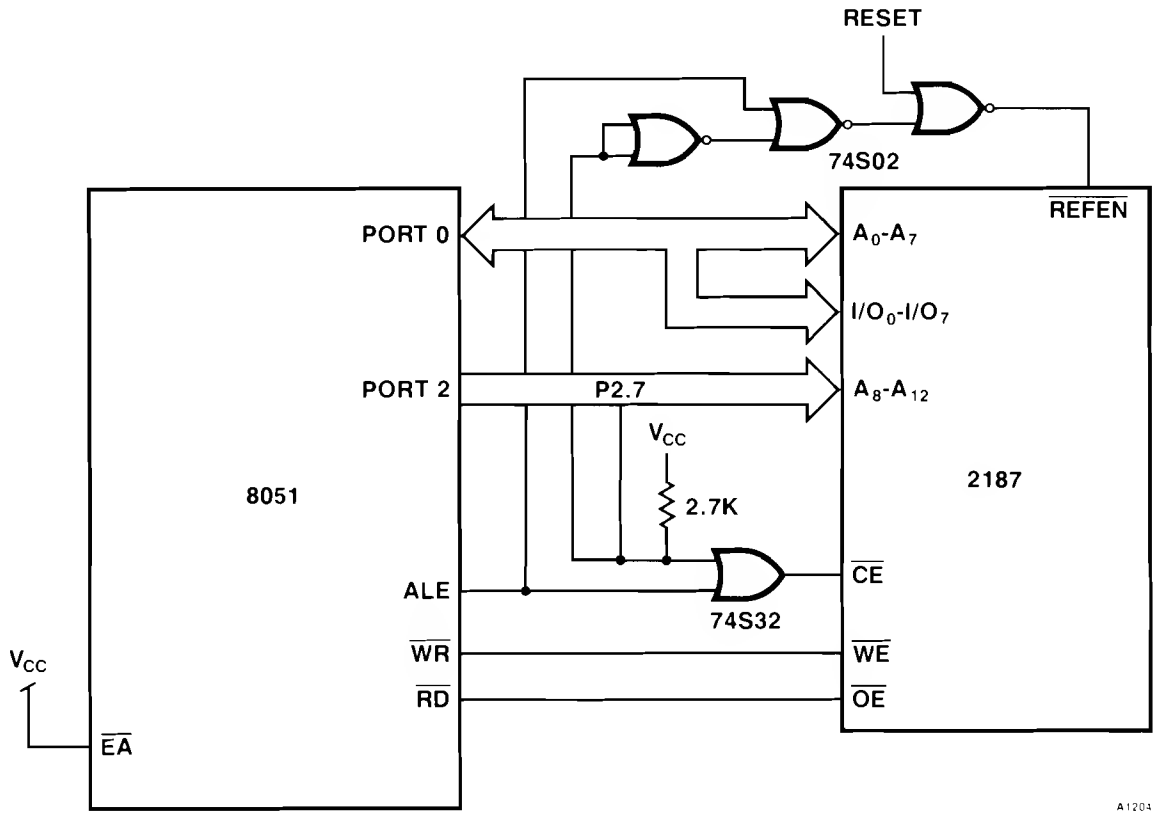


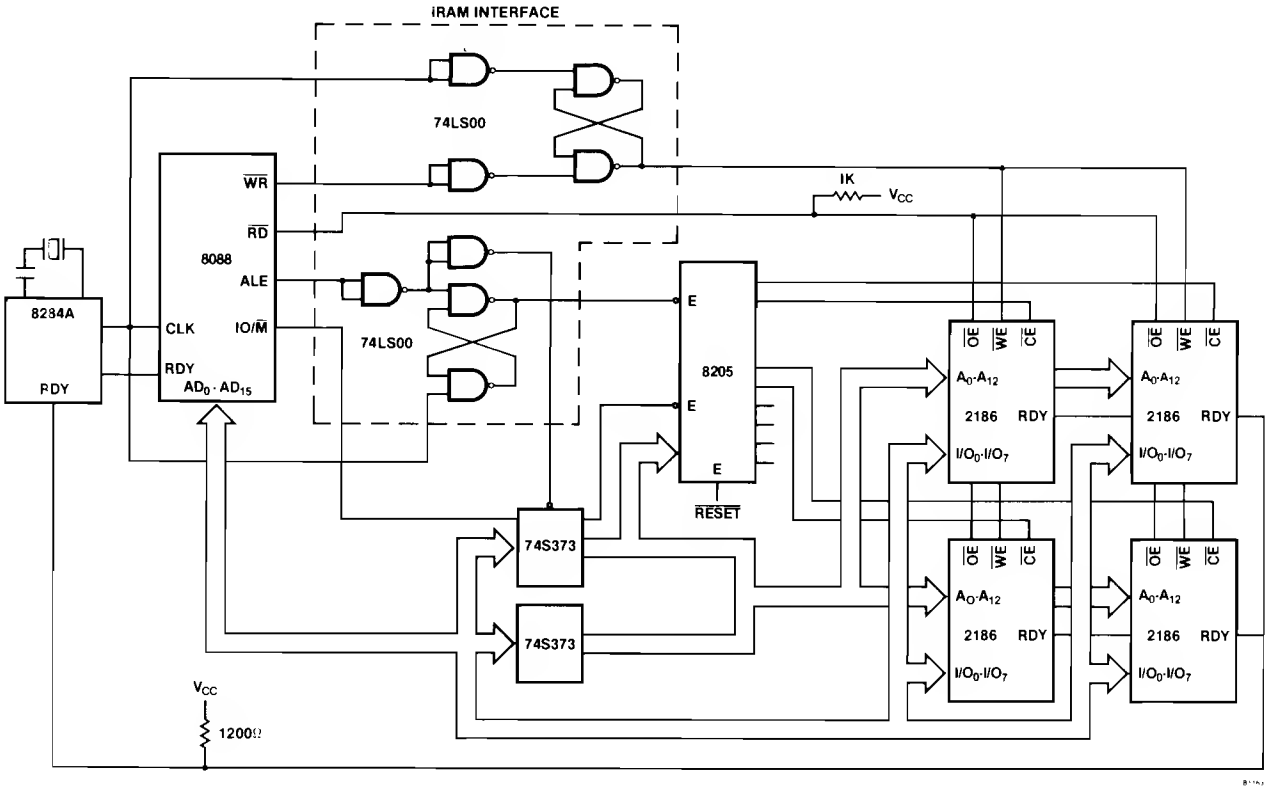
Figure 2. Delayed $\overline{WE}$

8051 MICROCONTROLLER SYSTEM



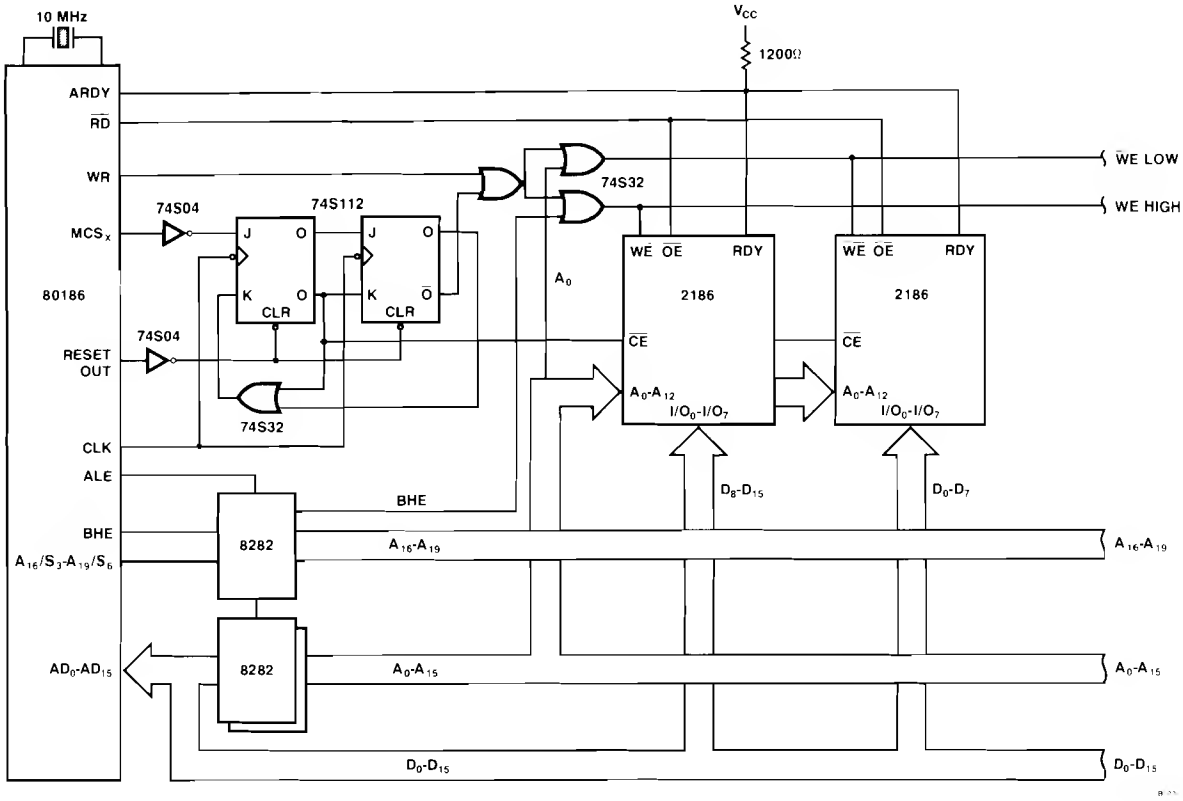
- USES THE 2187 SYNCHRONOUS REFRESH iRAM
- THE 2187 PROVIDES 8K BYTES OF EXTERNAL DATA MEMORY
- VERY SIMPLE INTERFACE
- NO ADDRESS LATCHES REQUIRED
- REFRESH OCCURS DURING INTERNAL MICROCONTROLLER OPCODE FETCHES
- SYSTEM RUNS AT 12 MHz

iAPX 88 MINIMUM MODE SYSTEM



- SIMPLE MINIMUM MODE SYSTEM
- FOUR 2186 iRAMs PROVIDE 32K BYTES OF LOCAL STORAGE
- CLEAN $\overline{CE}$ GENERATED BY ONE TTL PACKAGE (74LS00)
- ONE TTL PACKAGE (74LS00) DELAYS $\overline{WE}$ UNTIL DATA IS VALID
- THE iRAMs ARE IN UNIVERAL SITE SOCKETS — COMPATIBLE WITH SRAMs AND EPROMS
- SYSTEM RUNS AT 5 MHz WITH NO WAIT STATES

iAPX 186 SYSTEM



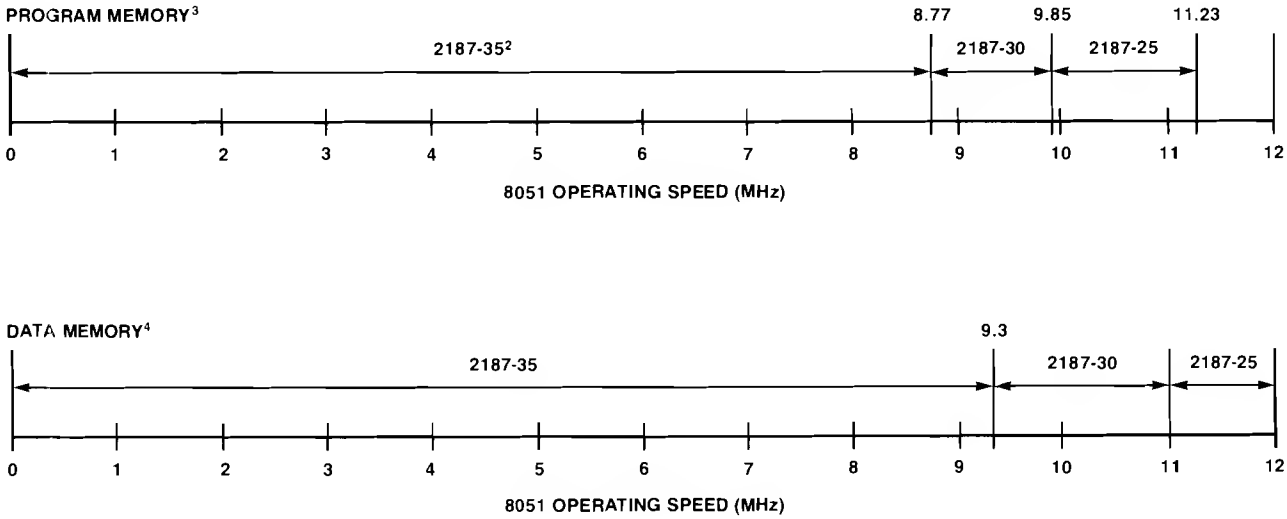
- **TWO 2186 iRAMs PROVIDE 8K WORDS OF LOCAL STORAGE**
- **SIMPLE CIRCUITRY GENERATES A CLEAN $\overline{CE}$**
- **DELAYED $\overline{WE}$ GATED TO SELECT ONE OR BOTH iRAMs FOR 8 OR 16 BIT DATA TRANSFERS**
- **READY HANDSHAKE LINE (RDY) CONNECTS DIRECTLY TO THE PROCESSOR**
- **THE iRAMs UNIVERSAL SITE SOCKETS ARE COMPATIBLE WITH SRAMs AND EPROMs**

Microprocessor Operating Speed Vs. iRAM Access Time[1]

Microprocessor Speed		0 Wait States	1 Wait State	2 Wait States	3 Wait States
8085AH	3 MHz	note 2	2186-35	2186-35	2186-35
	5 MHz		2186-35	2186-35	2186-35
	6 MHz		2186-30	2186-35	2186-35
8086[3]	5 Mhz	2186-30	2186-35	2186-35	2186-35
	8 MHz		2186-30	2186-35	2186-35
	10 MHz		2186-25	2186-35	2186-35
80186[4]	8 MHz		2186-30	2186-35	2186-35
	10 MHz			2186-30	2186-35
	15 MHz				2186-25
80286[4]	8 MHz			2186-30	2186-35
	10 MHz			2186-25	2186-35
	15 MHz				

NOTES:
1. Buffer delays not included.
2. Due to its RDY response requirements, the 8085 cannot run without wait states.
3. Timing also applicable for the 8088 microprocessor.
4. Specifications for higher clock speeds not available. Memory requirements for 10 MHz and 15 MHz are extrapolated from 8 MHz specifications.

8051 iRAM REQUIREMENTS[1]



NOTES:
1. Buffers delays not included.
2. Runs with no wait states over this speed range
3. The iRAM is used to store both program code and data.
4. Program code resides in the 8051's internal ROM or in external ROM.

B1218

2186/2187 Literature

Title	Description	For Whom
Data Sheets	2186S, 2187	All iRAM users
Designer's Guide to the iRAM	General introduction to iRAM system design, includes system block diagrams and timing matrix: system speed vs. iRAM access time.	All iRAM users
Designing memory systems with the 8K x 8 iRAM	Application Note 132 — describes the design of iRAM memory systems, includes a description of internal iRAM functions.	All iRAM users
The iRAM in microcontroller systems	How to design microcontroller memory systems using the synchronous 2187 iRAM. Available February 83.	All 2187 users
Smart Memories	Article Reprint 235 — overview of latest trends in smart memory technologies	All iRAM users
Memory Components Handbook	Data sheets and Ap Notes for all Intel Memory Components — includes 2186S, 2187 data sheets and AP 132	All iRAM users
iRAM System Reliability	A summary of the system testing performed on the 2186. Includes reliability data and test descriptions.	iRAM users who require extensive qualification
iRAM Arbiter Reliability	Report on the reliability of the iRAM arbitration circuit.	iRAM users who require extensive qualification

2186/2187 Literature Guide by Topic

	Topic				
Title	iRAM Tech. Design	2186 System Design	2187 System Design	iRAM Reliability	Order Number
2186S Data Sheet	★	★			210857-001
2187 Data Sheet	★		★		210859-001
AP-132 Designing Systems with iRAMs	★	★	★		210443-001
AR-235 Smart Memories	★				210784-001
Designer's Guide to iRAMs		★	★		210860
iRAM System Reliability	★			★	Available on Request
iRAM Arbiter Reliability	★			★	Available on Request
The iRAM in microcontroller systems	★		★		Available Feb. 83

Designer Notes

Designer Notes

INTEL CORPORATION, 3065 Bowers Avenue, Santa Clara, CA 95051 • (408) 987-8080

Printed in U.S.A./E-301/1282/10K/BA/AP